Universidad Popular del Cesar

# Minería de datos y Patrones

Version 2024-I

## HoG - Human Detection

[ Capítulo 2 ]

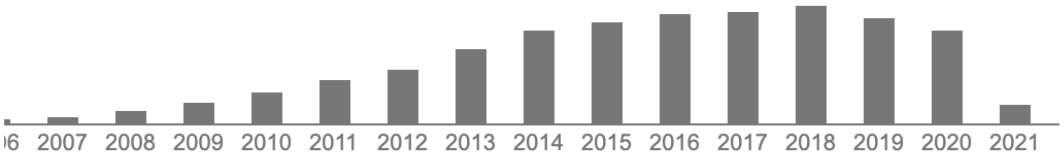
Dr. José Ramón Iglesias
DSP-ASIC BUILDER GROUP
Director Semillero TRIAC
Ingenieria Electronica
Universidad Popular del Cesar

# HoG: Histogram of oriented gradients

Navneet Dalal

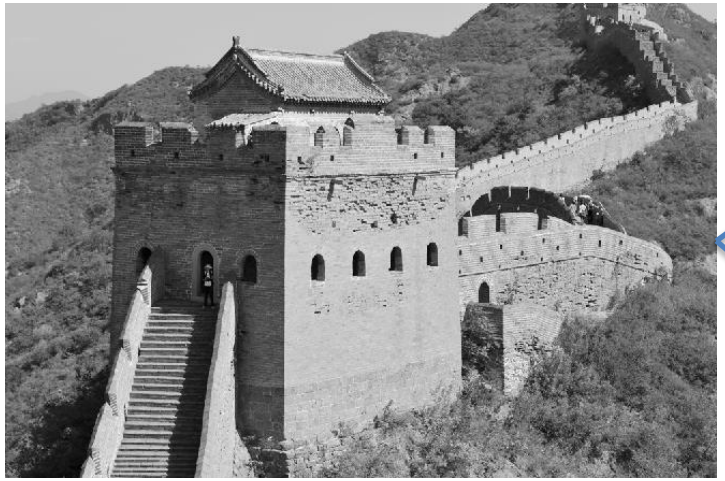
## Histograms of oriented gradients for human detection

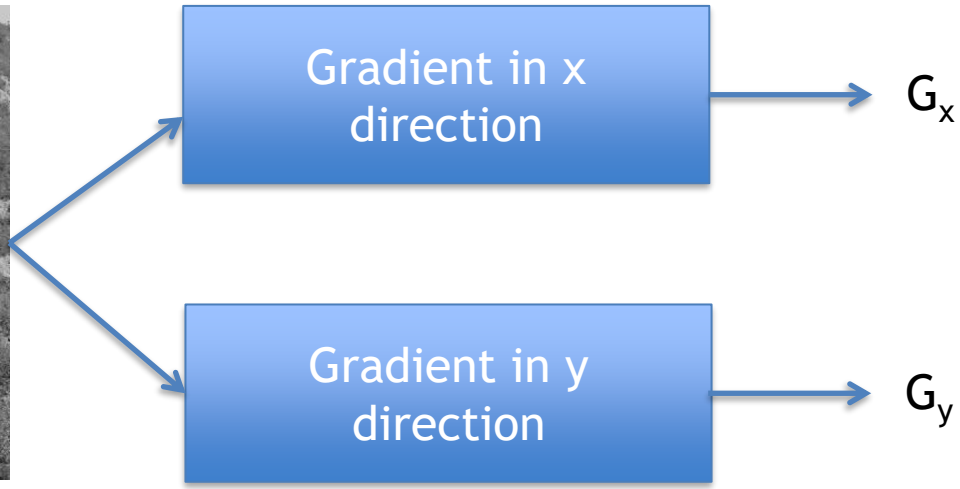
| | |
|---|---|
| Authors | Navneet Dalal, Bill Triggs |
| Publication date | 2005/6/20 |
| Conference | 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05) |
| Volume | 1 |
| Pages | 886-893 |
| Publisher | Ieee |
| Description | We study the question of feature sets for robust visual object recognition; adopting linear SVM based human detection as a test case. After reviewing existing edge and gradient based descriptors, we show experimentally that grids of histograms of oriented gradient (HOG) descriptors significantly outperform existing feature sets for human detection. We study the influence of each stage of the computation on performance, concluding that fine-scale gradients, fine orientation binning, relatively coarse spatial binning, and high-quality local contrast normalization in overlapping descriptor blocks are all important for good results. The new approach gives near-perfect separation on the original MIT pedestrian database, so we introduce a more challenging dataset containing over 1800 annotated human images with a large range of pose variations and backgrounds. |
| Total citations | Cited by 35511 |




N Dalal, B Triggs: Histograms of oriented gradients for human detection. Computer Vision and Pattern Recognition, (CVPR 2005).

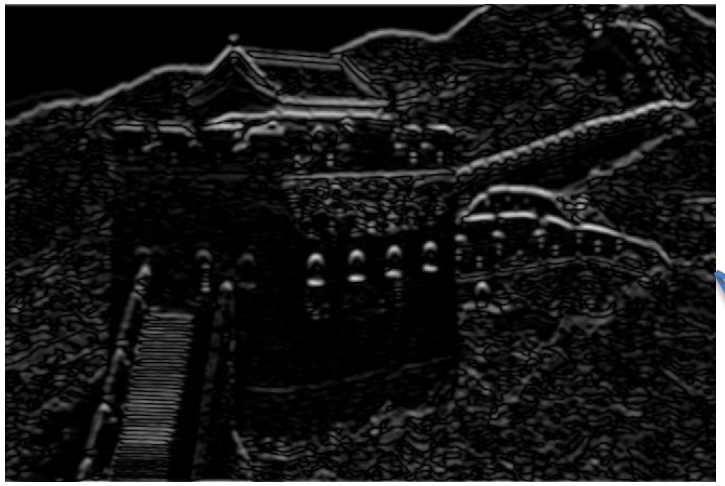
# Histogram of Gradients



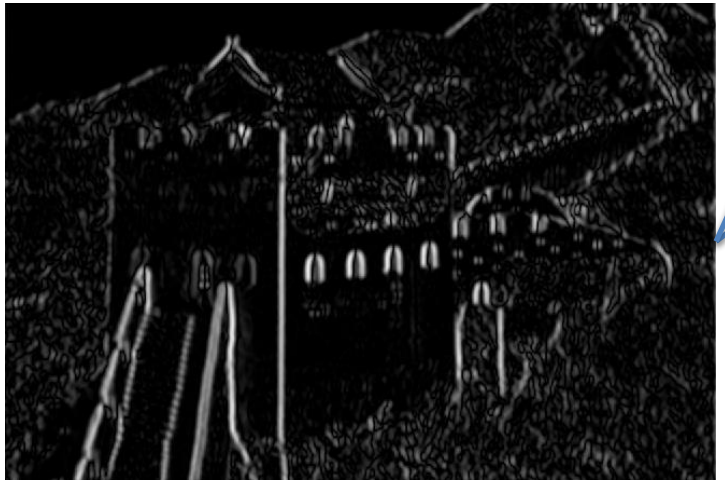


# Histogram of Gradients

$G_x$: Gradient in x direction



$G_y$: Gradient in y direction



Magnitude → R

$= \text{sqrt}(G_x^2+G_y^2)$

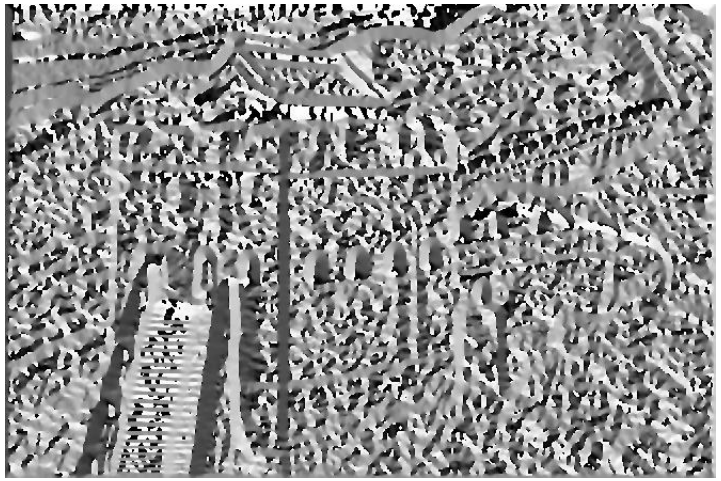
Angle → A

$= \arctan(Gy/G_x)$

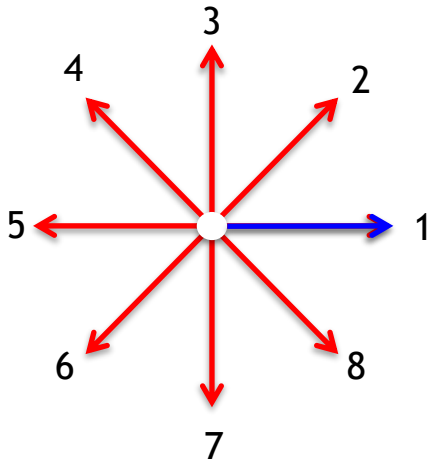
# Histogram of Gradients

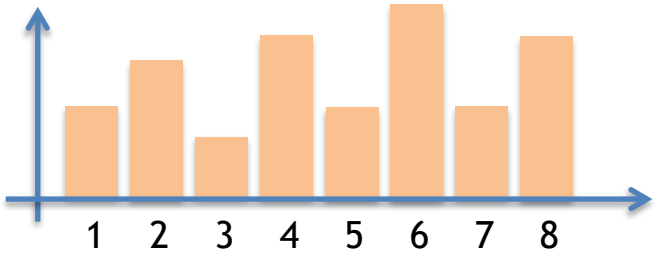
R: Magnitude

A: Angle



Histogram of 8 directions

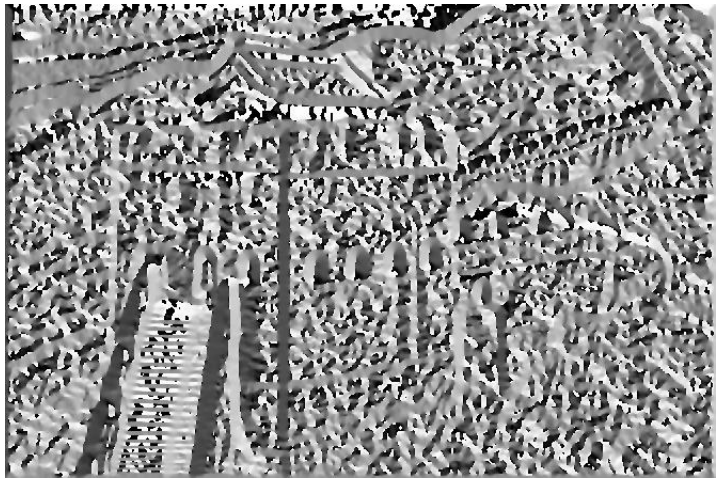


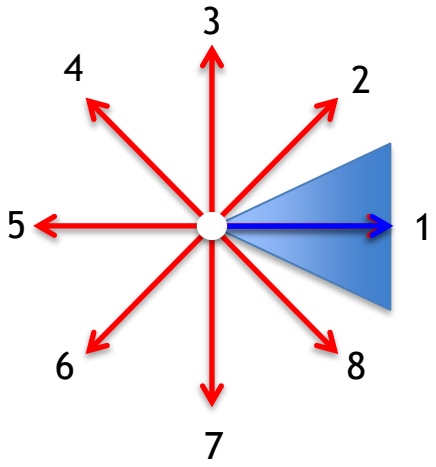
# Histogram of Gradients

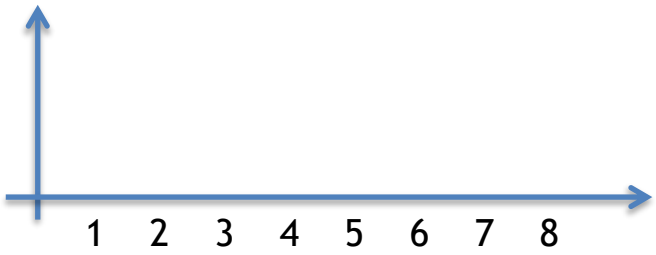
R: Magnitude

A: Angle



Histogram of 8 directions
(computation of first bin)

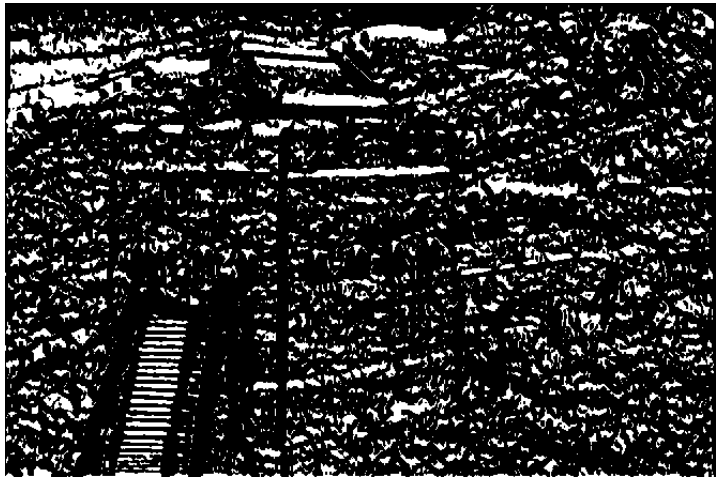


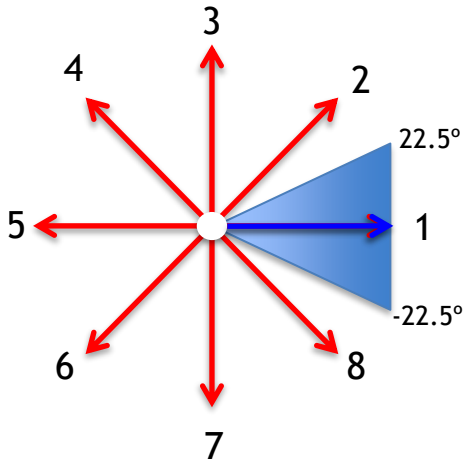
# Histogram of Gradients

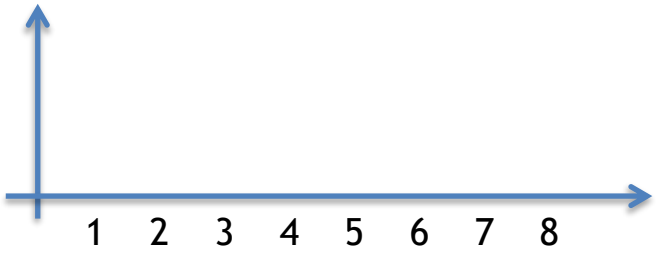
R: Magnitude

A: Angle between -22.5° and 22.5°



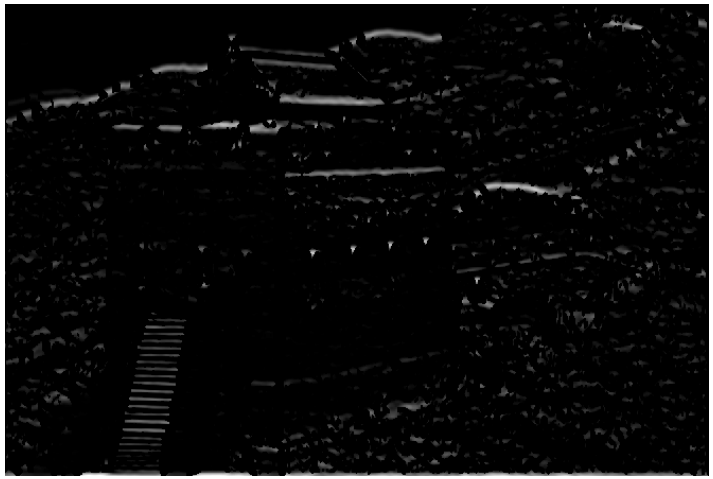
Histogram of 8 directions
(computation of first bin)



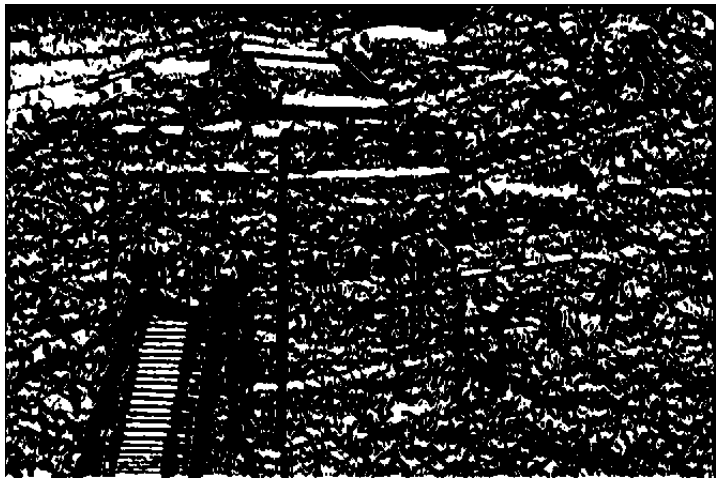



# Histogram of Gradients

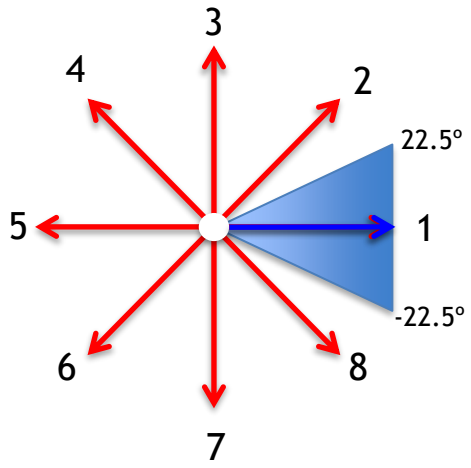
R: Magnitude in this direction



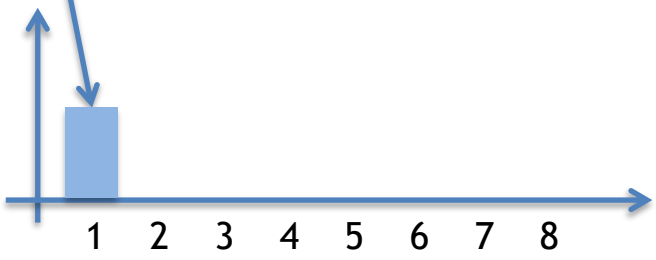
A: Angle between -22.5° and 22.5°



Histogram of 8 directions
(computation of first bin)

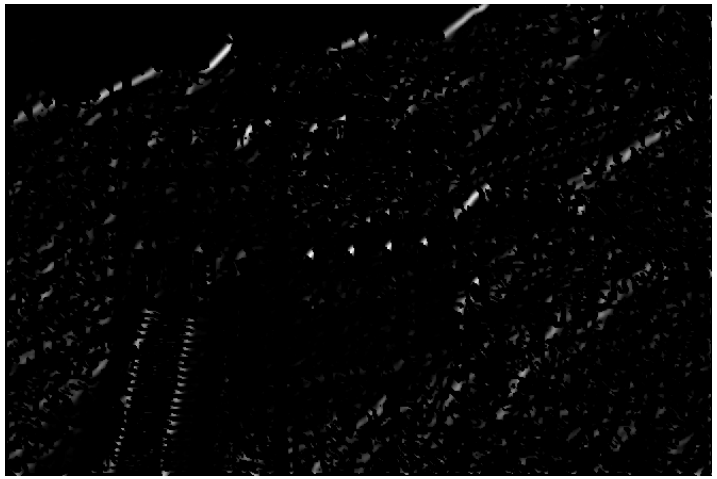



Σ



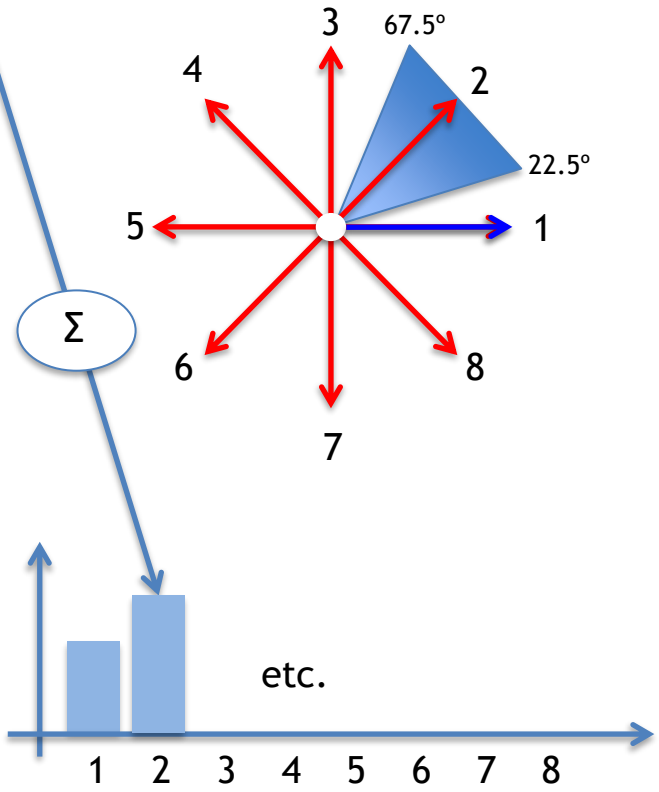
# Histogram of Gradients

R: Magnitude in this direction



A: Angle between 22.5° and 67.5°

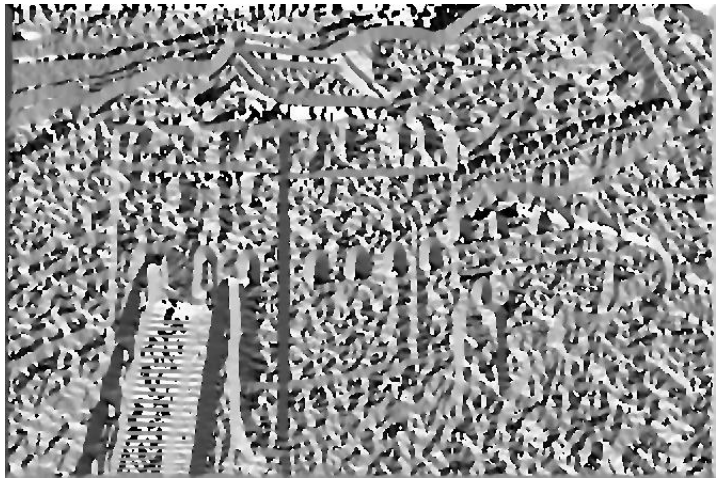
Histogram of 8 directions
(computation of second bin)



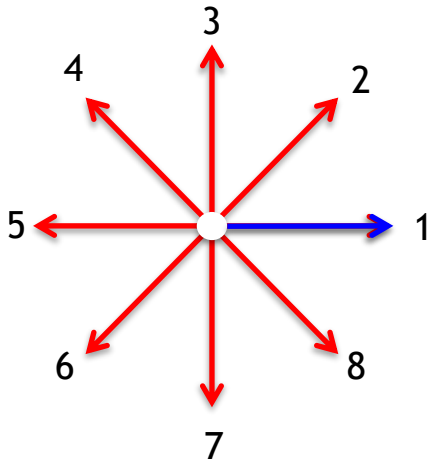
# Histogram of Gradients

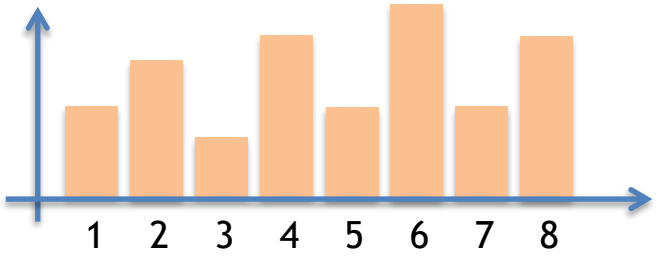
R: Magnitude

A: Angle



Histogram of 8 directions

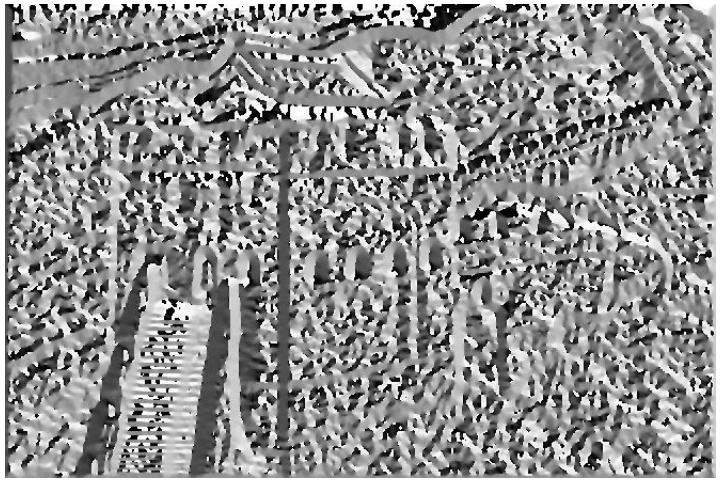


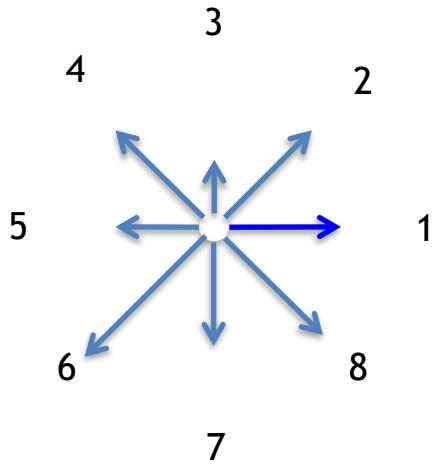
# Histogram of Gradients

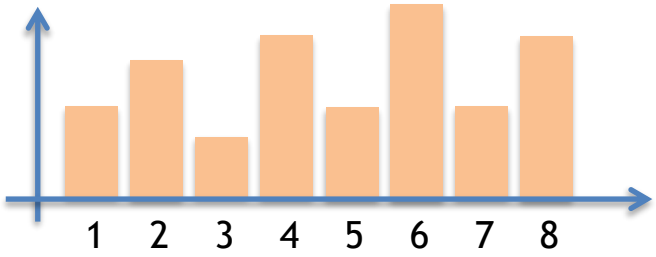
R: Magnitude

A: Angle

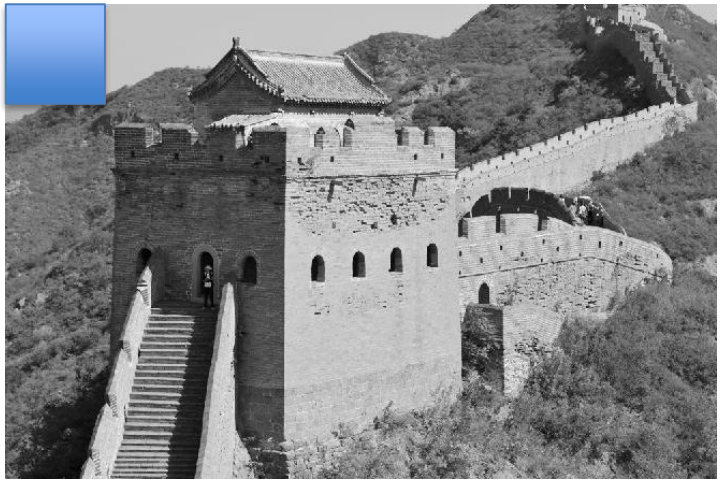


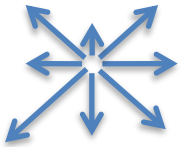
Histogram of 8 directions

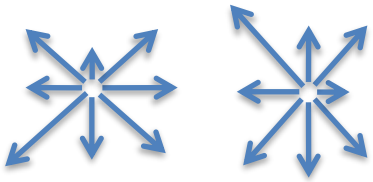


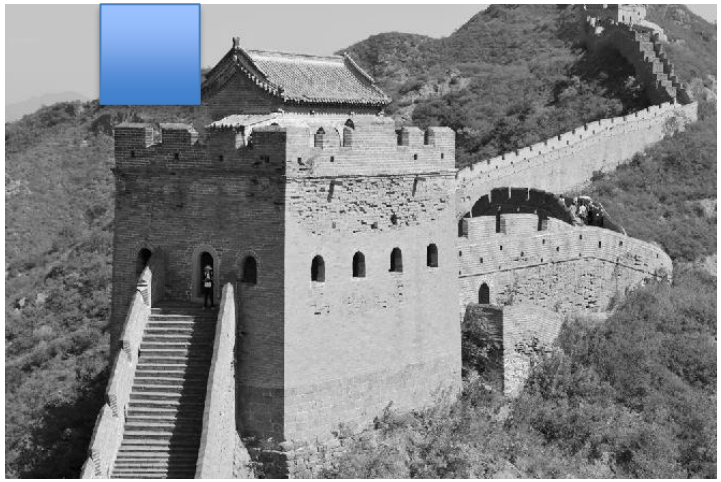



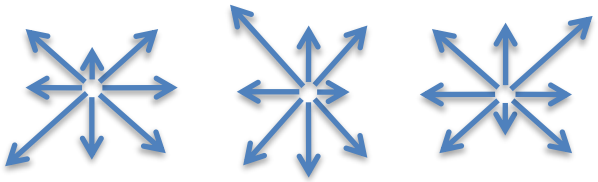
The descriptor proposed by the authors is a concatenation of HoG in different overlapped partitions

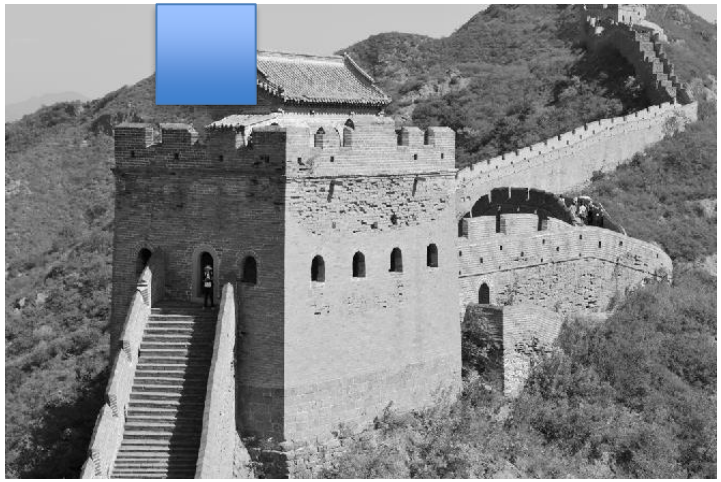


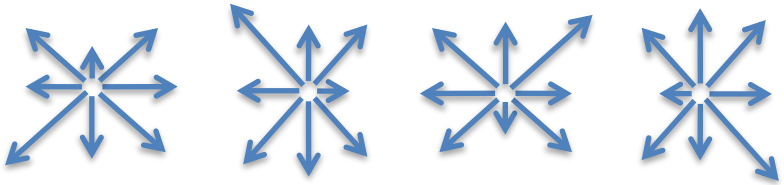


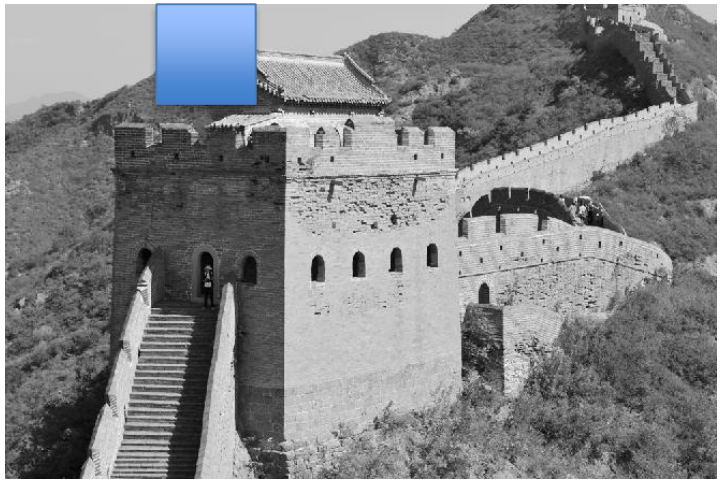


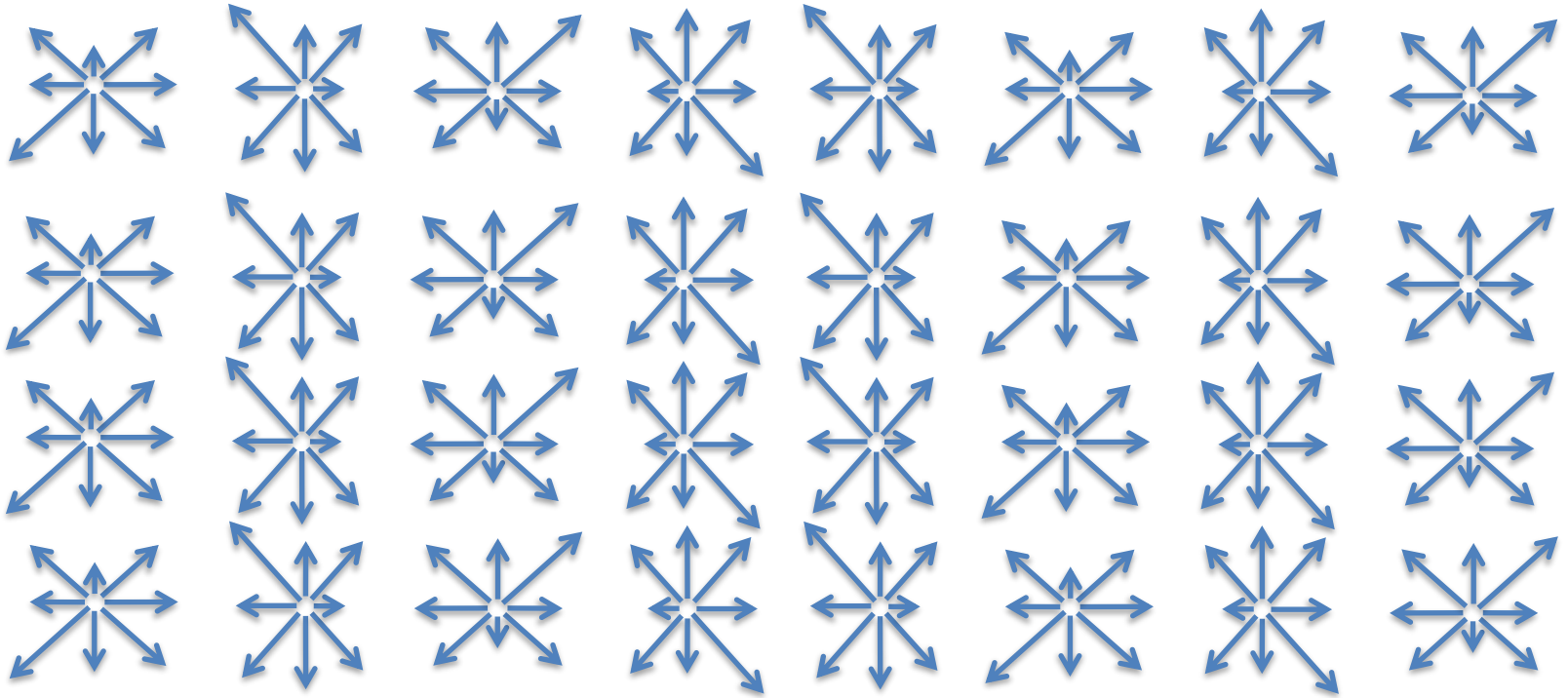


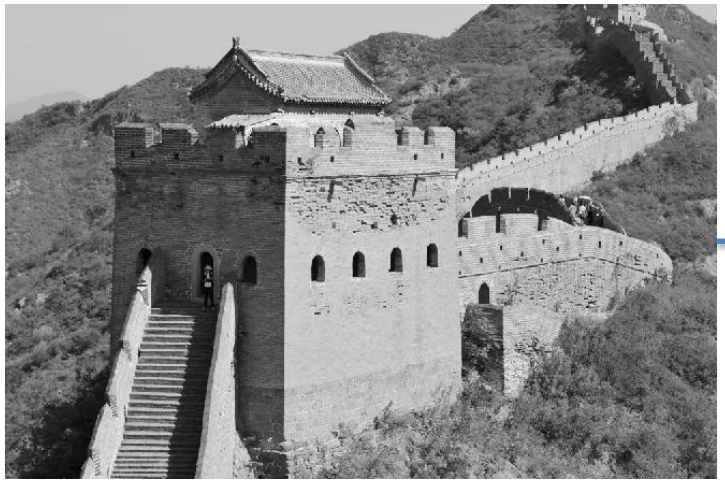




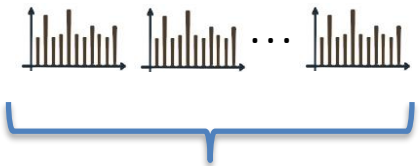






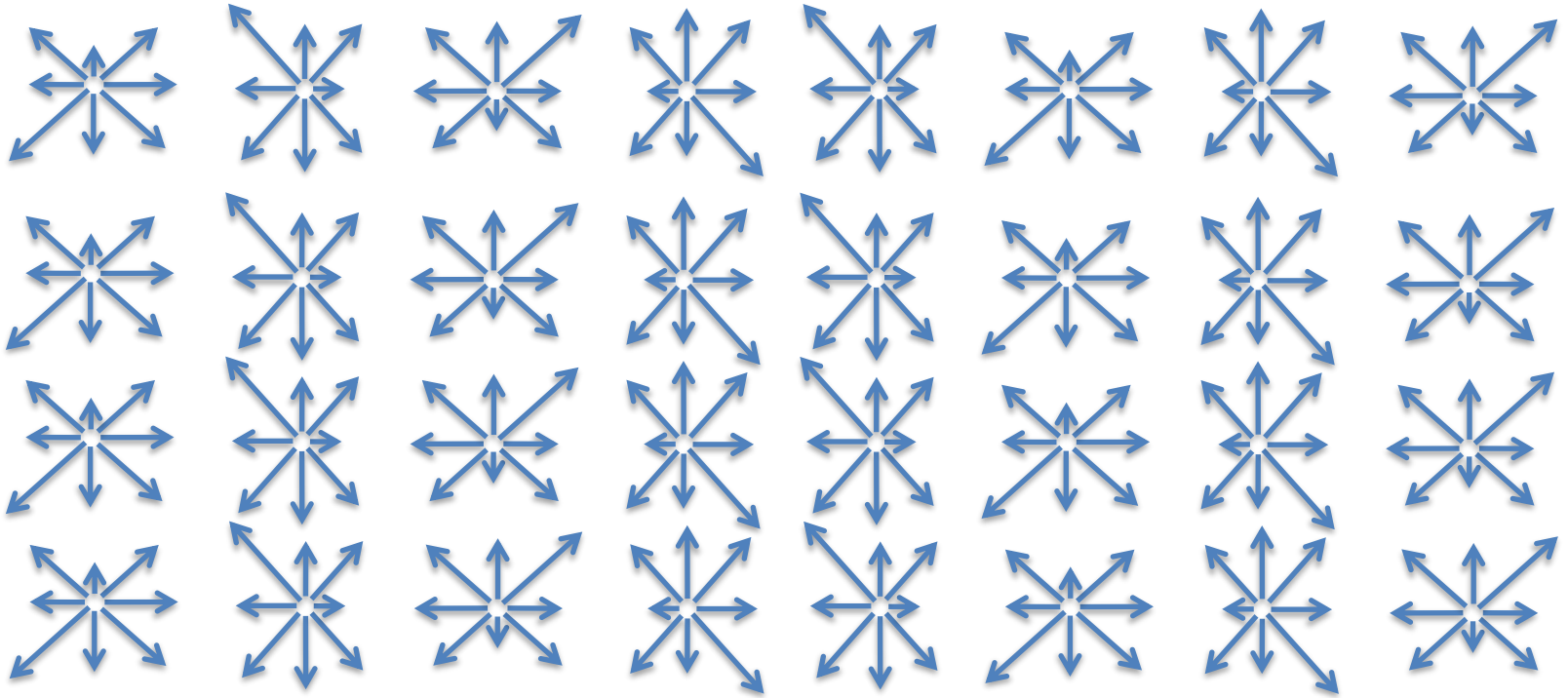


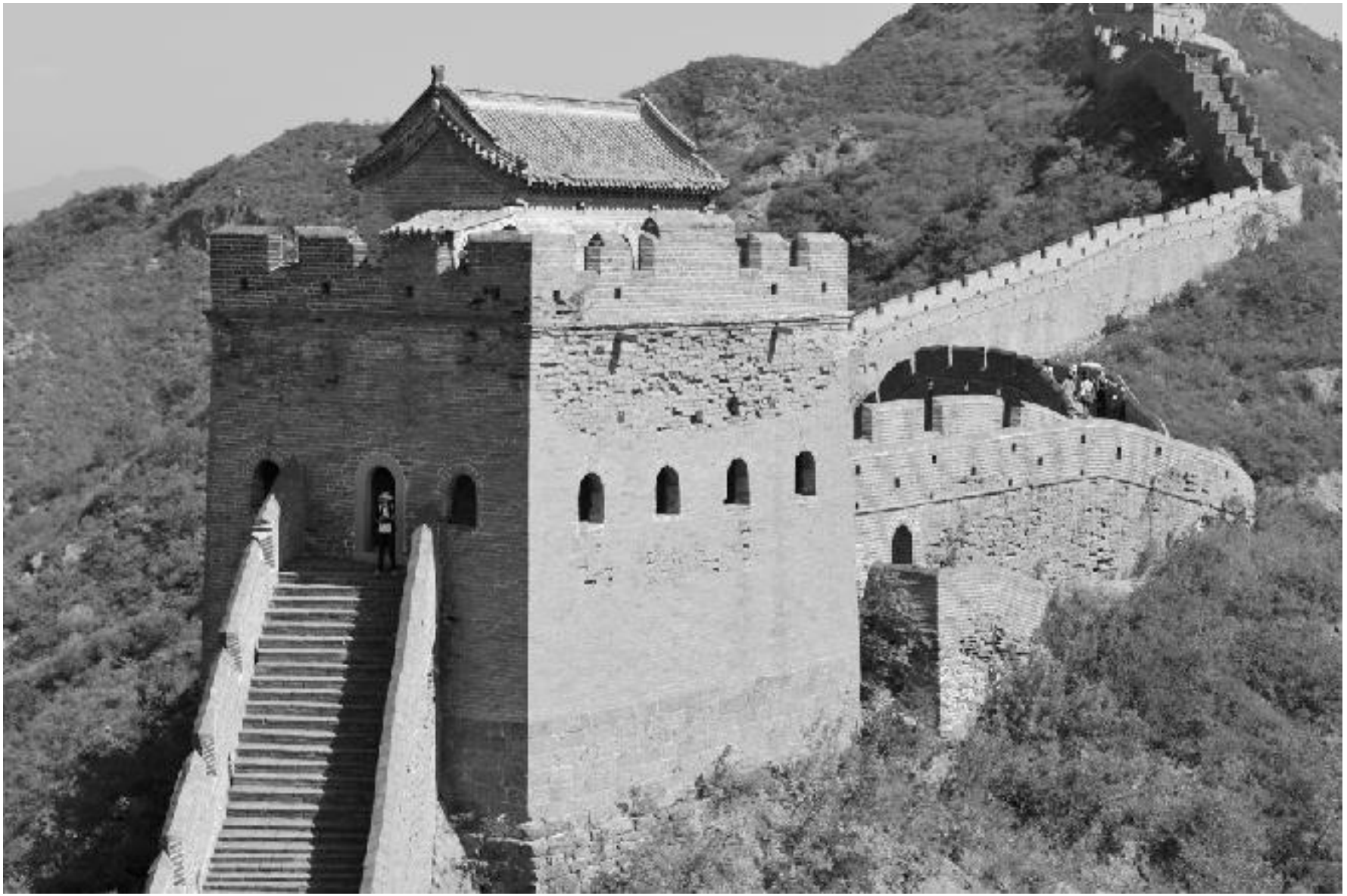
Feature Extraction

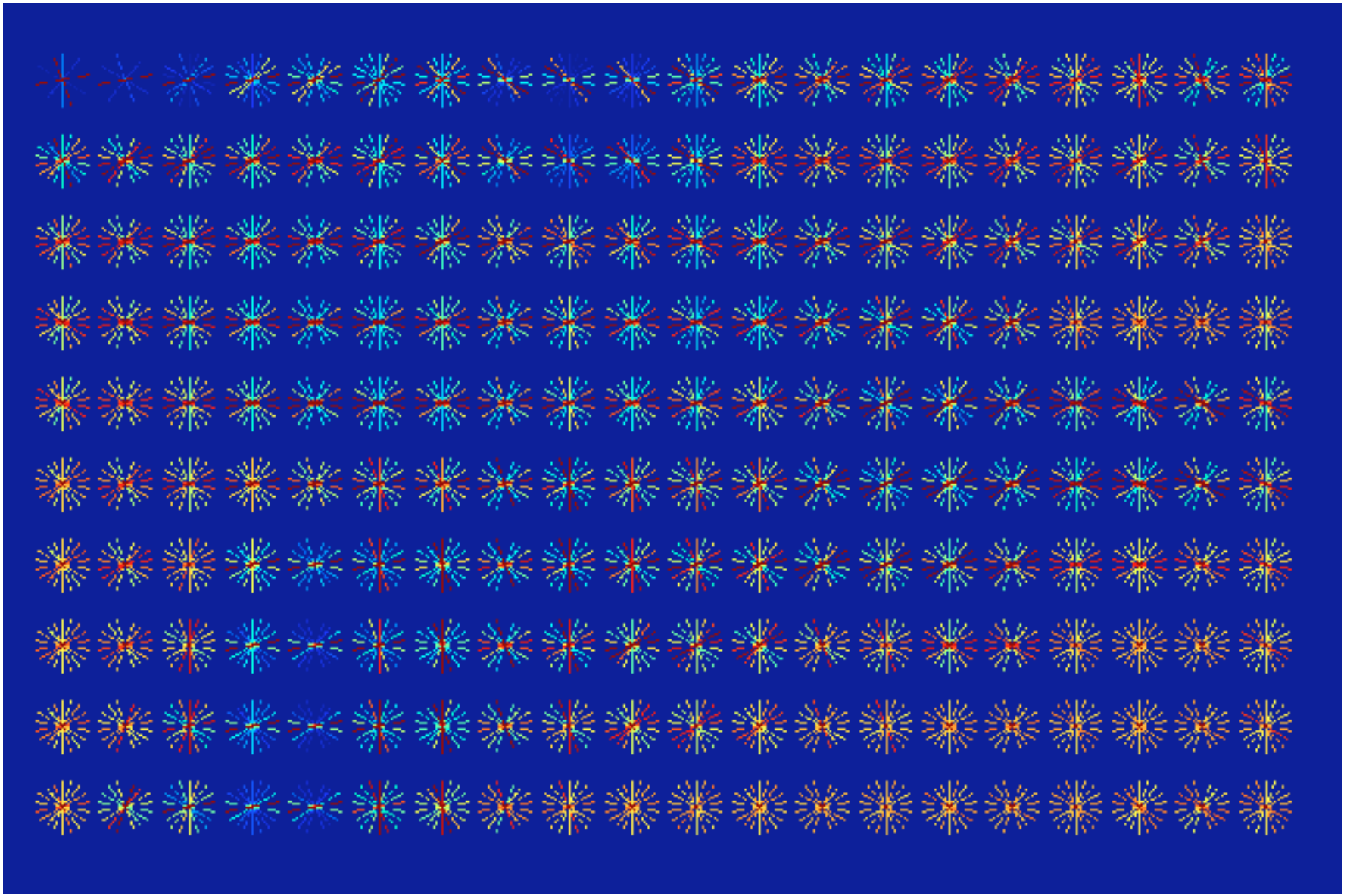


Descriptor of n elements

n = # cells x # bins







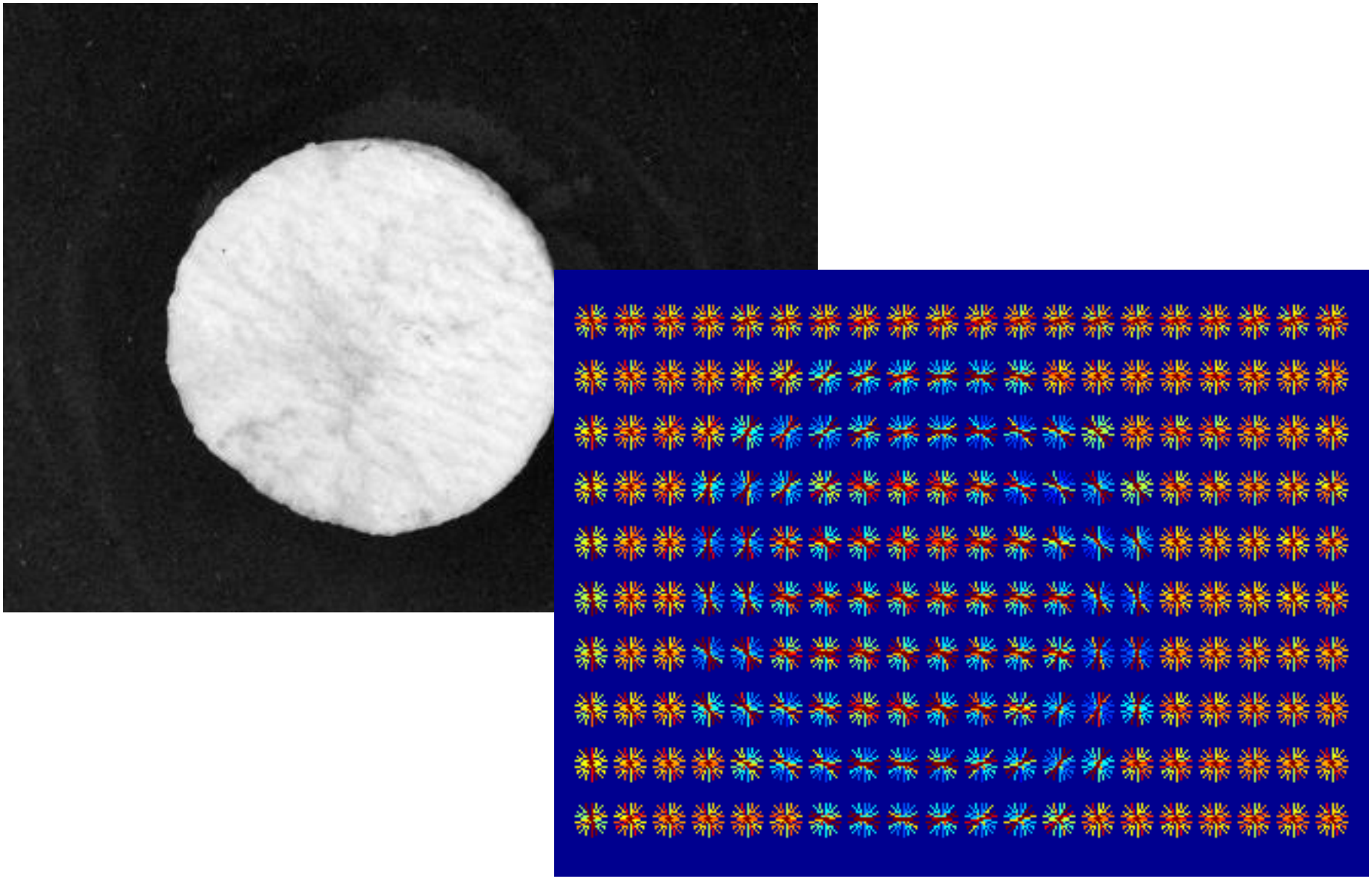
Example using Python:

```
def hog(img, orientations=9,
pixels_per_cell=(16,16),cells_per_block=(2,2),norm=False):
  X = skimage.feature.hog(img, orientations=orientations,
pixels_per_cell=pixels_per_cell,cells_per_block=cells_per_block)
  if norm:
    X = X/np.linalg.norm(X)
  return X
```

La función `hog()` toma 6 parámetros como entrada:

- `image`: la imagen de destino a la que desea aplicar la extracción de características HOG.
- `Orientaciones`: Número de contenedores en el histograma que queremos crear, el trabajo de investigación original utilizó 9 contenedores, por lo que pasaremos 9 como orientaciones.
- `pixels_per_cell`: Determina el tamaño de la celda, como mencionamos anteriormente, es 8x8.
- `cells_per_block`: El número de celdas por bloque, será de 2x2 como se mencionó anteriormente.
- `visualizar`: Un booleano si para devolver la imagen del HOG, la establecemos en `True` para que podamos mostrar la imagen.
- `multicanal`: Lo establecemos en `True` para indicar a la función que la última dimensión se considera como un canal de color, en lugar de espacial.

Example using Balu:



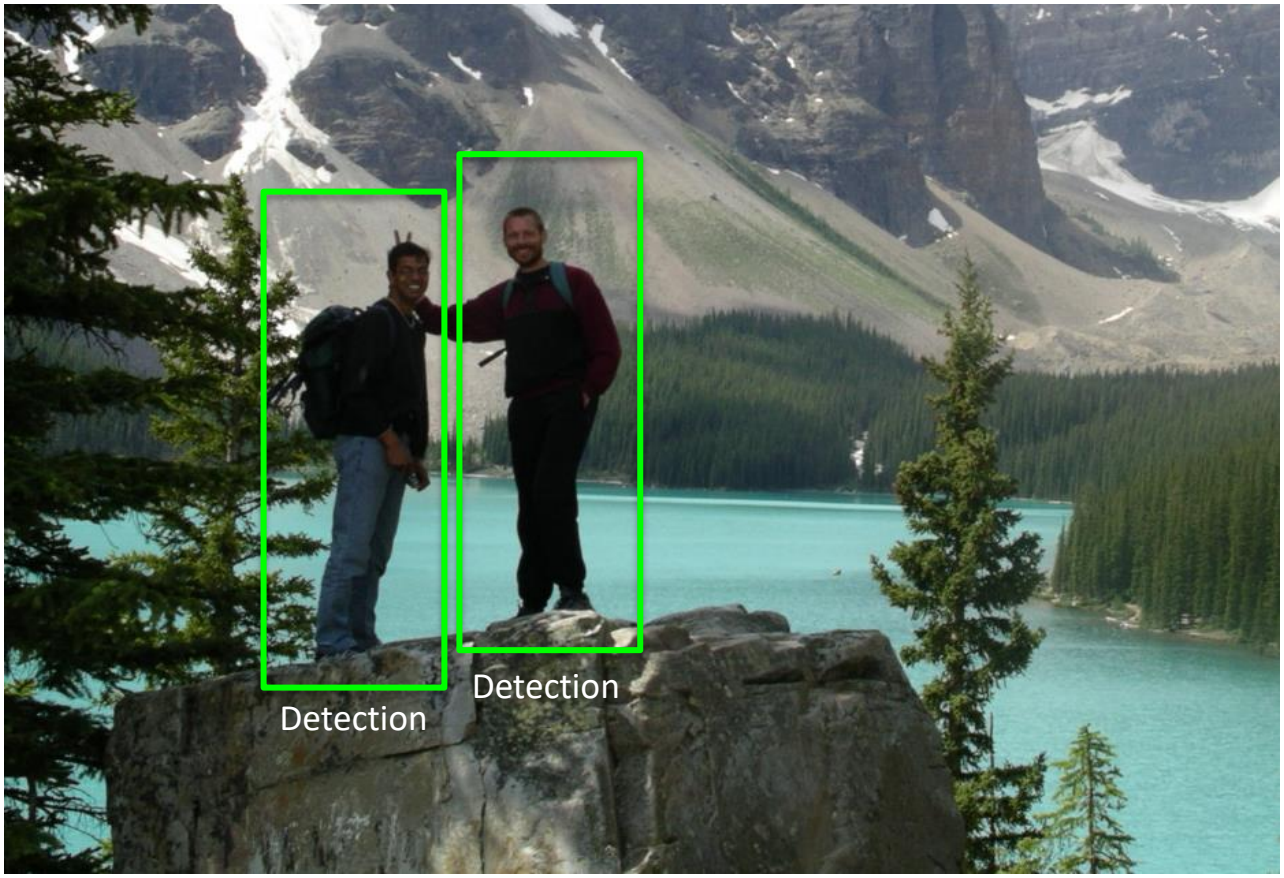
# How to detect pedestrians using HoG?

# The Solution

# Strategy: Sliding Windows

# Strategy: Sliding Windows

# Strategy: Sliding Windows



The Key-Idea:

Design a classifier that is able to distinguish between pedestrians from no-pedestrians.

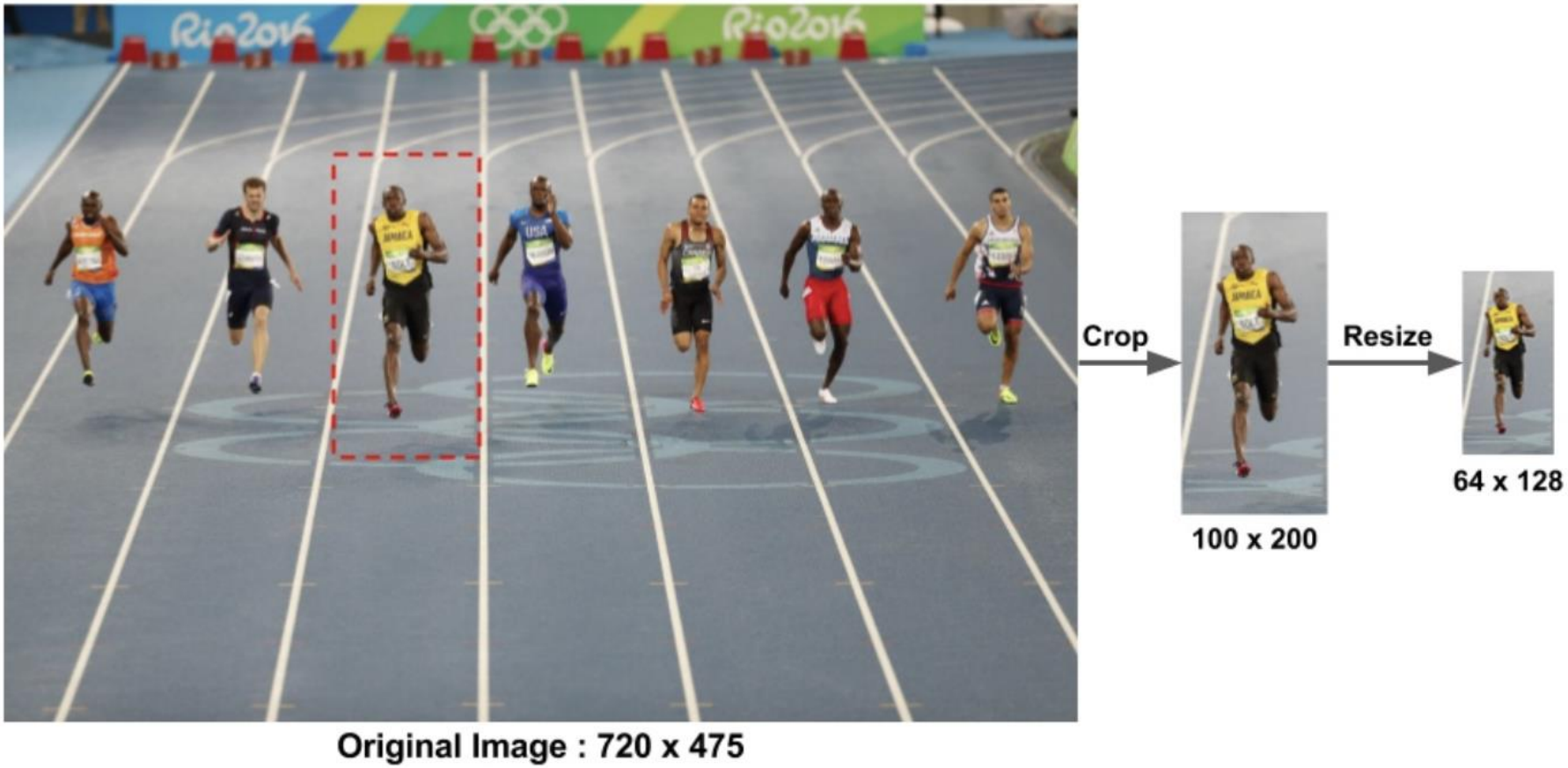
# Positive Class: Pedestrians

Pedestrian

# Negative Class: No-Pedestrians

No-Pedestrian

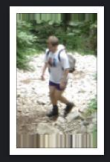
# Construction of the Dataset: Positive Class



# Positive Class: Pedestrians

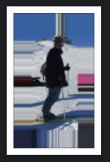

crop_000010a.png


crop_000010b.png


crop_000011a.png


crop_000011b.png


crop_000603a.png

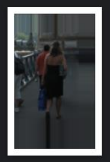

crop_000603b.png


crop_000606a.png

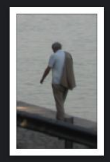

crop_000606b.png


crop_000606c.png


crop_000606d.png


crop_000607c.png

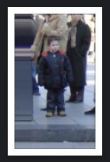

crop_000607d.png

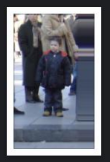

crop_000608a.png


crop_000608b.png


crop001001a.png


crop001001b.png


crop001001c.png


crop001001d.png


crop001001e.png


crop001001f.png


crop001002c.png


crop001002d.png


crop001003a.png


crop001003b.png


crop001003c.png


crop001003d.png


crop001004a.png


crop001004b.png


crop001004c.png


crop001004d.png


crop001005c.png


crop001005d.png


crop001006a.png

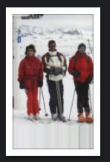

crop001006b.png

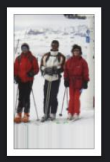

crop001006c.png


crop001006d.png


crop001007a.png


crop001007b.png


crop001007c.png


crop001007d.png


crop001007g.png


crop001007h.png


crop001007i.png


crop001007j.png


crop001008a.png


crop001008b.png


crop001008c.png

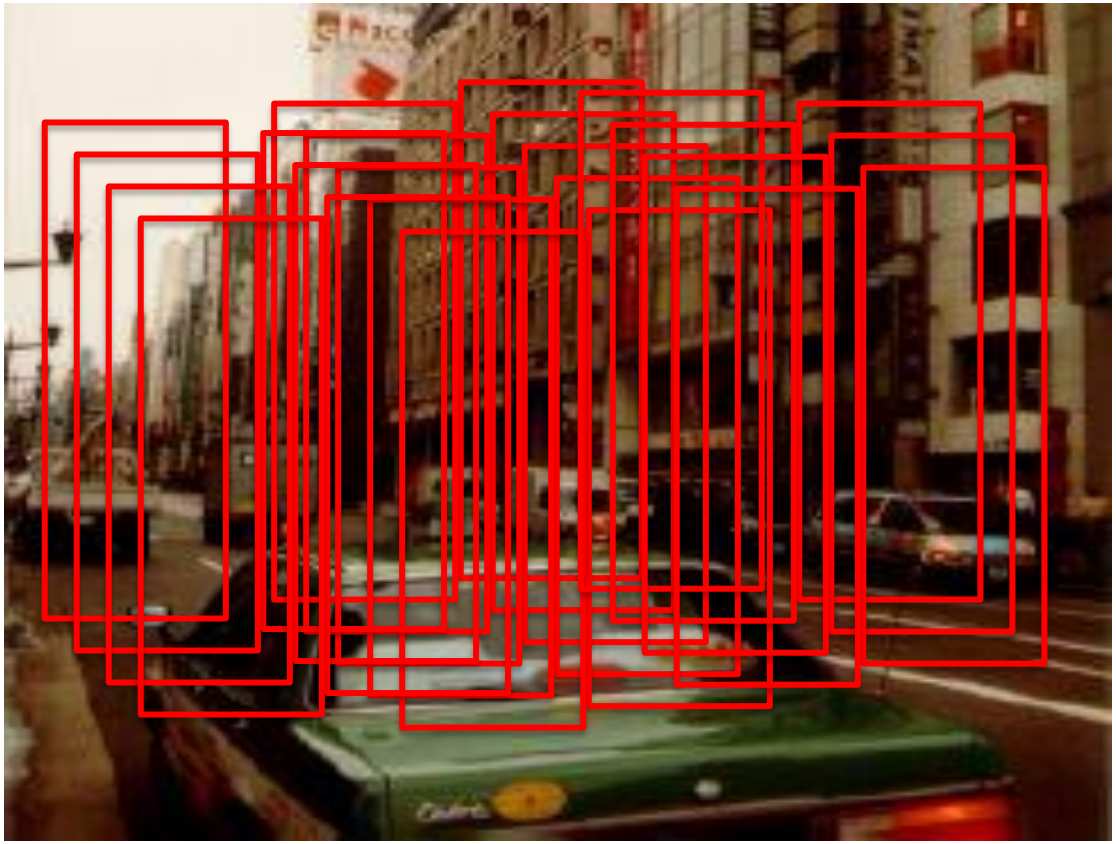

crop001008d.png


crop001009a.png


crop001009b.png

# Construction of the Dataset: Negative Class

# Construction of the Dataset: Negative Class



# Negative Class: No-Pedestrians

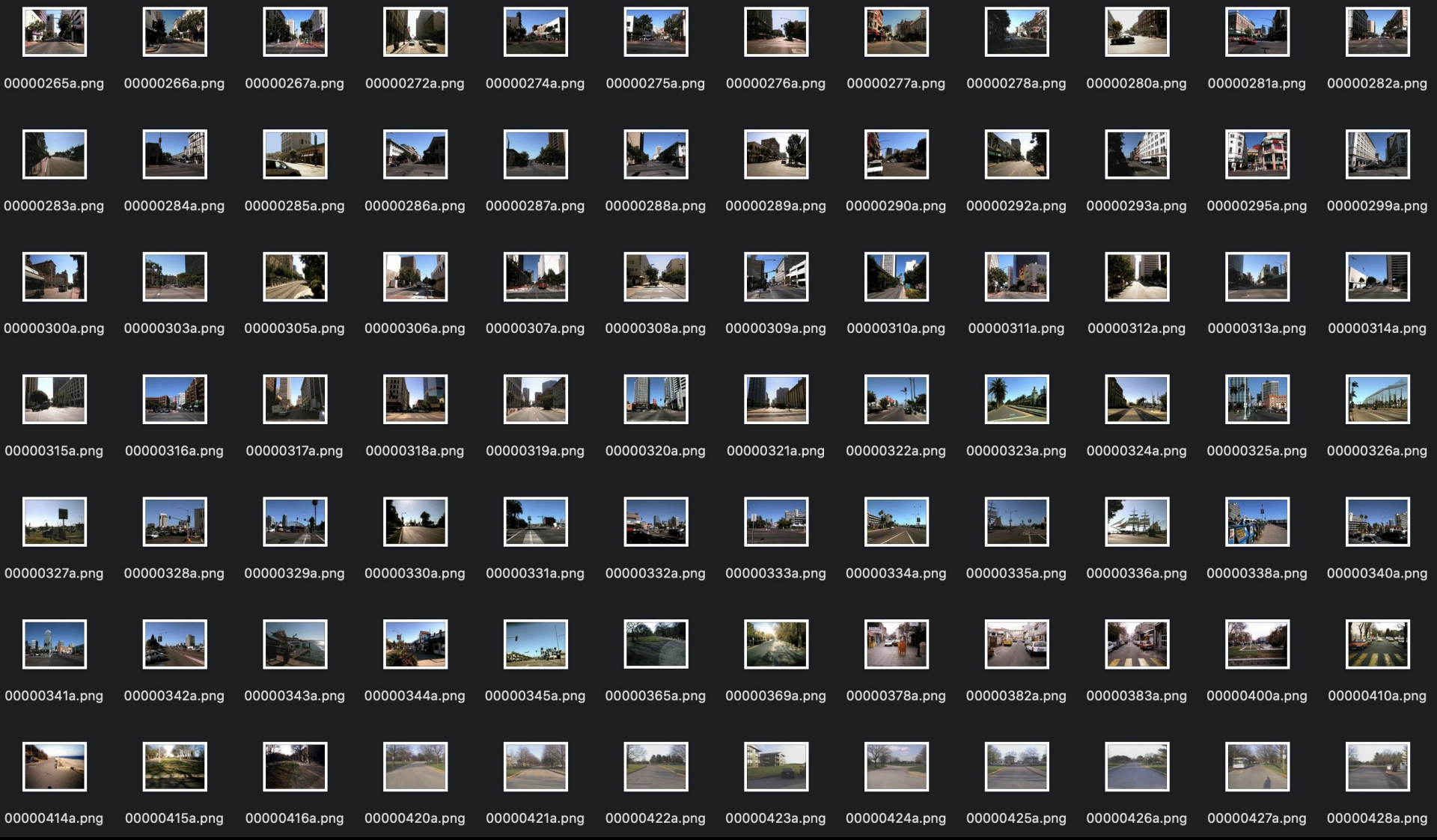
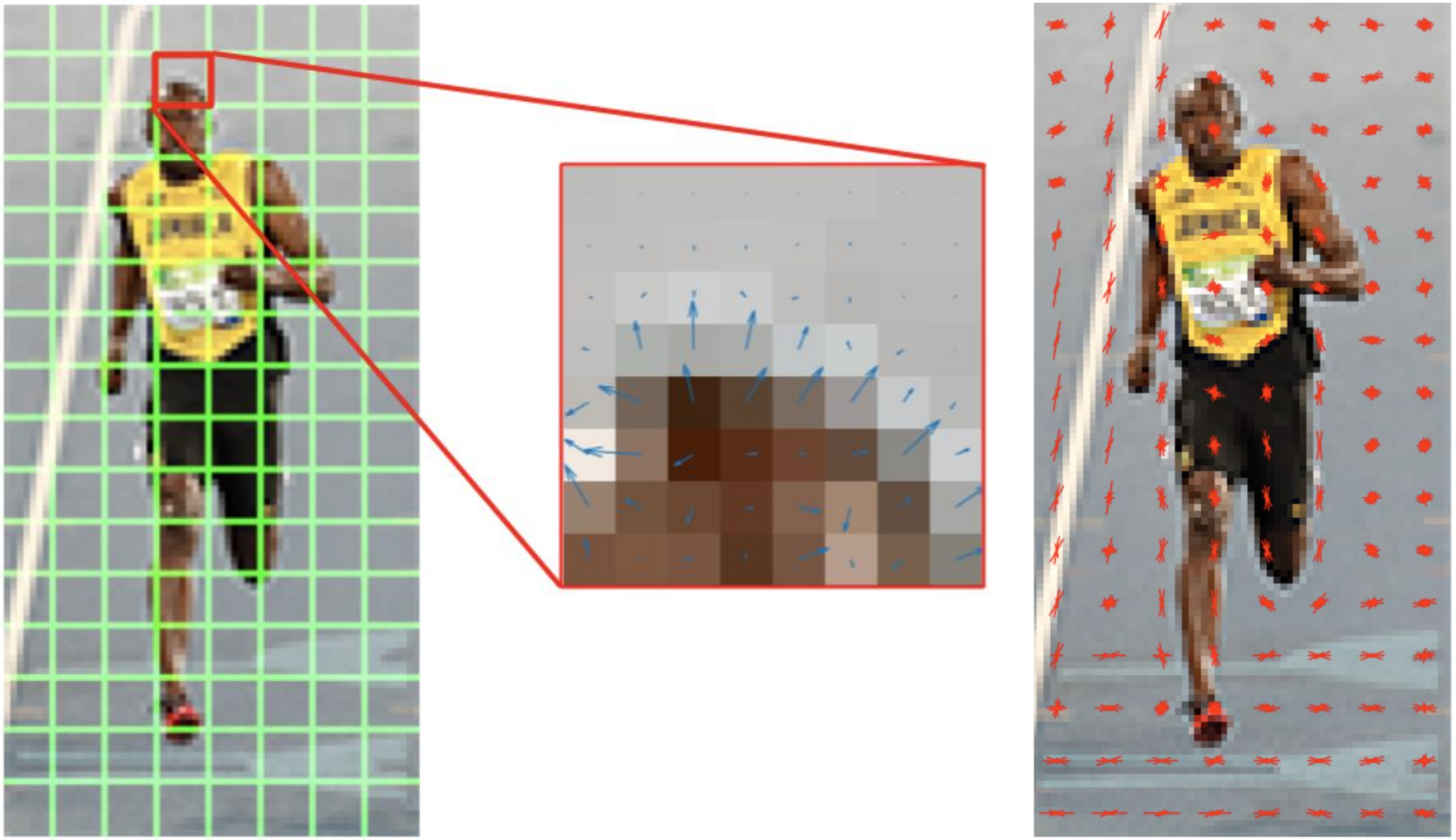


Image:
128 x 64 x 3 pixels

# Feature Extraction

Descriptor
3780 elements



**Center : The RGB patch and gradients represented using arrows. Right : The gradients in the same patch represented as numbers**

https://www.learnopencv.com/histogram-of-oriented-gradients/

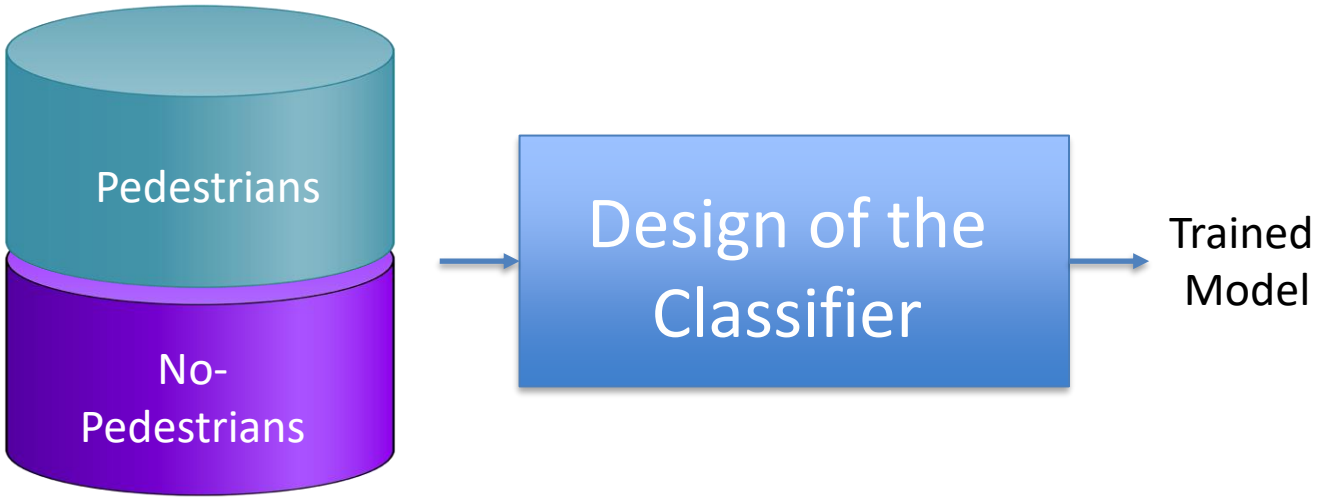
# Dataset: Positive Class


| Pedestrian | Element-1 | Element-2 | Element-3 | Element-4 | Element-5 | . . . | Element-3780 |
|---|---|---|---|---|---|---|---|
| 1 | 0.3452 | 0.1151 | 0.2685 | 0.1342 | 0.2416 | | 0.2301 |
| 2 | 0.2301 | 0.1534 | 0.2148 | 0.1611 | 0.2071 | | 0.3068 |
| 3 | 0.1726 | 0.1918 | 0.1790 | 0.1879 | 0.1812 | | 0.3836 |
| 4 | 0.1381 | 0.2301 | 0.1534 | 0.2148 | 0.1611 | | 0.4603 |
| 5 | 0.1151 | 0.2685 | 0.1342 | 0.2416 | 0.1450 | | 0.5370 |
| 6 | 0.0986 | 0.3068 | 0.1193 | 0.2685 | 0.1318 | | 0.6137 |
| 7 | 0.0863 | 0.3452 | 0.1074 | 0.2953 | 0.1208 | | 0.6904 |
| 8 | 0.0767 | 0.3836 | 0.0976 | 0.3222 | 0.1115 | | 0.7671 |
| 9 | 0.0690 | 0.4219 | 0.0895 | 0.3490 | 0.1036 | | 0.8438 |
| 10 | 0.0628 | 0.4603 | 0.0826 | 0.3759 | 0.0967 | | 0.9205 |
| 11 | 0.0575 | 0.4986 | 0.0767 | 0.4027 | 0.0906 | | 0.9972 |
| 12 | 0.0531 | 0.5370 | 0.0716 | 0.4296 | 0.0853 | | 1.0740 |
| 13 | 0.0493 | 0.5753 | 0.0671 | 0.4564 | 0.0805 | | 1.1507 |
| 14 | 0.0460 | 0.6137 | 0.0632 | 0.4833 | 0.0763 | | 1.2274 |
| 15 | 0.0432 | 0.6520 | 0.0597 | 0.5101 | 0.0725 | | 1.3041 |
| 16 | 0.0406 | 0.6904 | 0.0565 | 0.5370 | 0.0690 | | 1.3808 |
| 17 | 0.0384 | 0.7288 | 0.0537 | 0.5638 | 0.0659 | | 1.4575 |
| 18 | 0.0363 | 0.7671 | 0.0511 | 0.5907 | 0.0630 | | 1.5342 |
| 19 | 0.0345 | 0.8055 | 0.0488 | 0.6175 | 0.0604 | | 1.6109 |
| 20 | 0.0329 | 0.8438 | 0.0467 | 0.6444 | 0.0580 | | 1.6876 |
| 21 | 0.0314 | 0.8822 | 0.0447 | 0.6712 | 0.4851 | | 1.7644 |
| 22 | 0.0300 | 0.9205 | 0.0430 | 0.0802 | 0.6444 | | 1.8411 |
| 23 | 0.0288 | 0.9589 | 0.3593 | 0.0604 | 0.4660 | | 1.9178 |
| 24 | 0.0276 | 0.1146 | 0.4773 | 0.0835 | 0.3625 | | 0.2293 |
| 25 | 0.2310 | 0.0863 | 0.3452 | 0.1074 | 0.2953 | | 0.1726 |
| : | : | : | : | : | : | | : |
| : | : | : | : | : | : | | : |
| 100000 | 0.1230 | 0.0493 | 0.5753 | 0.0671 | 0.4564 | | 0.0986 |

# Dataset: Negative Class


| No-Pedestrian | Element-1 | Element-2 | Element-3 | Element-4 | Element-5 | . . . | Element-3780 |
|---|---|---|---|---|---|---|---|
| 1 | 0.7452 | 0.2484 | 0.5796 | 0.2898 | 0.5216 | | 0.4968 |
| 2 | 0.4968 | 0.3312 | 0.4637 | 0.3478 | 0.4471 | | 0.6624 |
| 3 | 0.3726 | 0.4140 | 0.3864 | 0.4057 | 0.3912 | | 0.8280 |
| 4 | 0.2981 | 0.4968 | 0.3312 | 0.4637 | 0.3478 | | 0.9936 |
| 5 | 0.2484 | 0.5796 | 0.2898 | 0.5216 | 0.3130 | | 1.1592 |
| 6 | 0.2129 | 0.6624 | 0.2576 | 0.5796 | 0.2845 | | 1.3248 |
| 7 | 0.1863 | 0.7452 | 0.2318 | 0.6376 | 0.2608 | | 1.4904 |
| 8 | 0.1656 | 0.8280 | 0.2108 | 0.6955 | 0.2408 | | 1.6560 |
| 9 | 0.1490 | 0.9108 | 0.1932 | 0.7535 | 0.2236 | | 1.8216 |
| 10 | 0.1355 | 0.9936 | 0.1783 | 0.8114 | 0.2087 | | 1.9872 |
| 11 | 0.1242 | 1.0764 | 0.1656 | 0.8694 | 0.1956 | | 2.1528 |
| 12 | 0.1146 | 1.1592 | 0.1546 | 0.9274 | 0.1841 | | 2.3184 |
| 13 | 0.1065 | 1.2420 | 0.1449 | 0.9853 | 0.1739 | | 2.4840 |
| 14 | 0.0994 | 1.3248 | 0.1364 | 1.0433 | 0.1647 | | 2.6496 |
| 15 | 0.0932 | 1.4076 | 0.1288 | 1.1012 | 0.1565 | | 2.8152 |
| 16 | 0.0877 | 1.4904 | 0.1220 | 1.1592 | 0.1490 | | 2.9808 |
| 17 | 0.0828 | 1.5732 | 0.1159 | 1.2172 | 0.1423 | | 3.1464 |
| 18 | 0.0784 | 1.6560 | 0.1104 | 1.2751 | 0.1361 | | 3.3120 |
| 19 | 0.0745 | 1.7388 | 0.1054 | 1.3331 | 0.1304 | | 3.4776 |
| 20 | 0.0710 | 1.8216 | 0.1008 | 1.3910 | 0.1252 | | 3.6432 |
| 21 | 0.0677 | 1.9044 | 0.0966 | 1.4490 | 0.4851 | | 3.8088 |
| 22 | 0.0648 | 1.9872 | 0.0927 | 0.3740 | 1.3910 | | 3.9744 |
| 23 | 0.0621 | 2.0700 | 0.3593 | 0.1304 | 1.0060 | | 4.1400 |
| 24 | 0.0596 | 0.5342 | 1.0304 | 0.1803 | 0.7825 | | 1.0684 |
| 25 | 0.2310 | 0.1863 | 0.7452 | 0.2318 | 0.6376 | | 0.3726 |
| : | : | : | : | : | : | | : |
| : | : | : | : | : | : | | : |
| 800000 | 0.9230 | 0.1065 | 1.2420 | 0.1449 | 0.9853 | | 0.2129 |

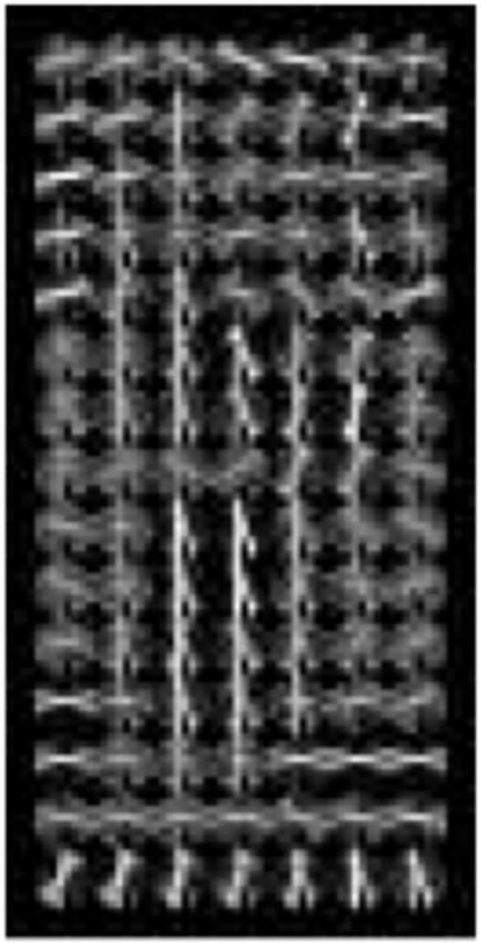
Training




Dataset

Image:
128 x 64 x 3 pixels

Descriptor
3780 elements



Output:
0 / 1

Pedestrian

Feature Extraction
*Using HoG Features*

Classification
*Using Trained Model*


Example

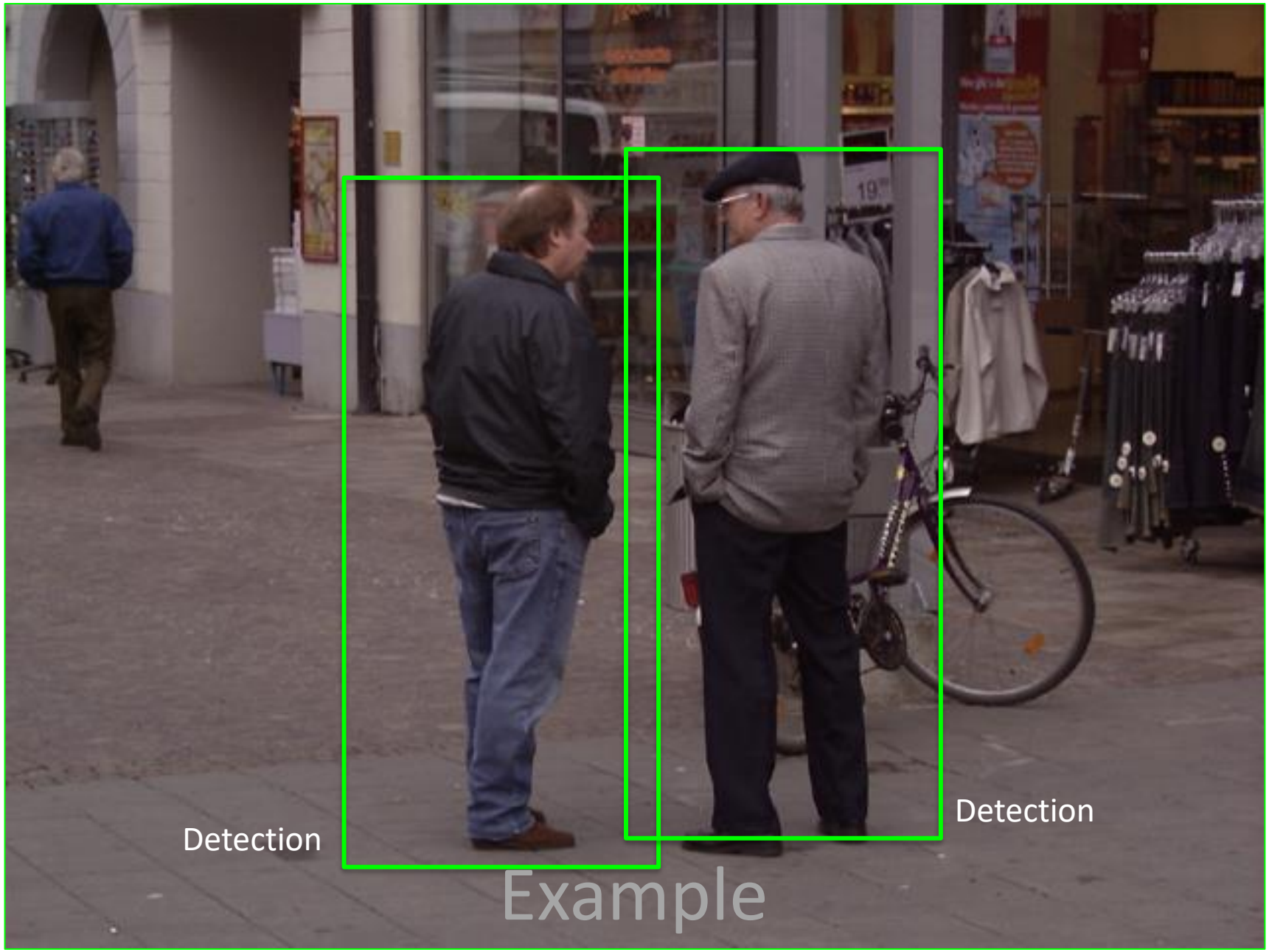

Example

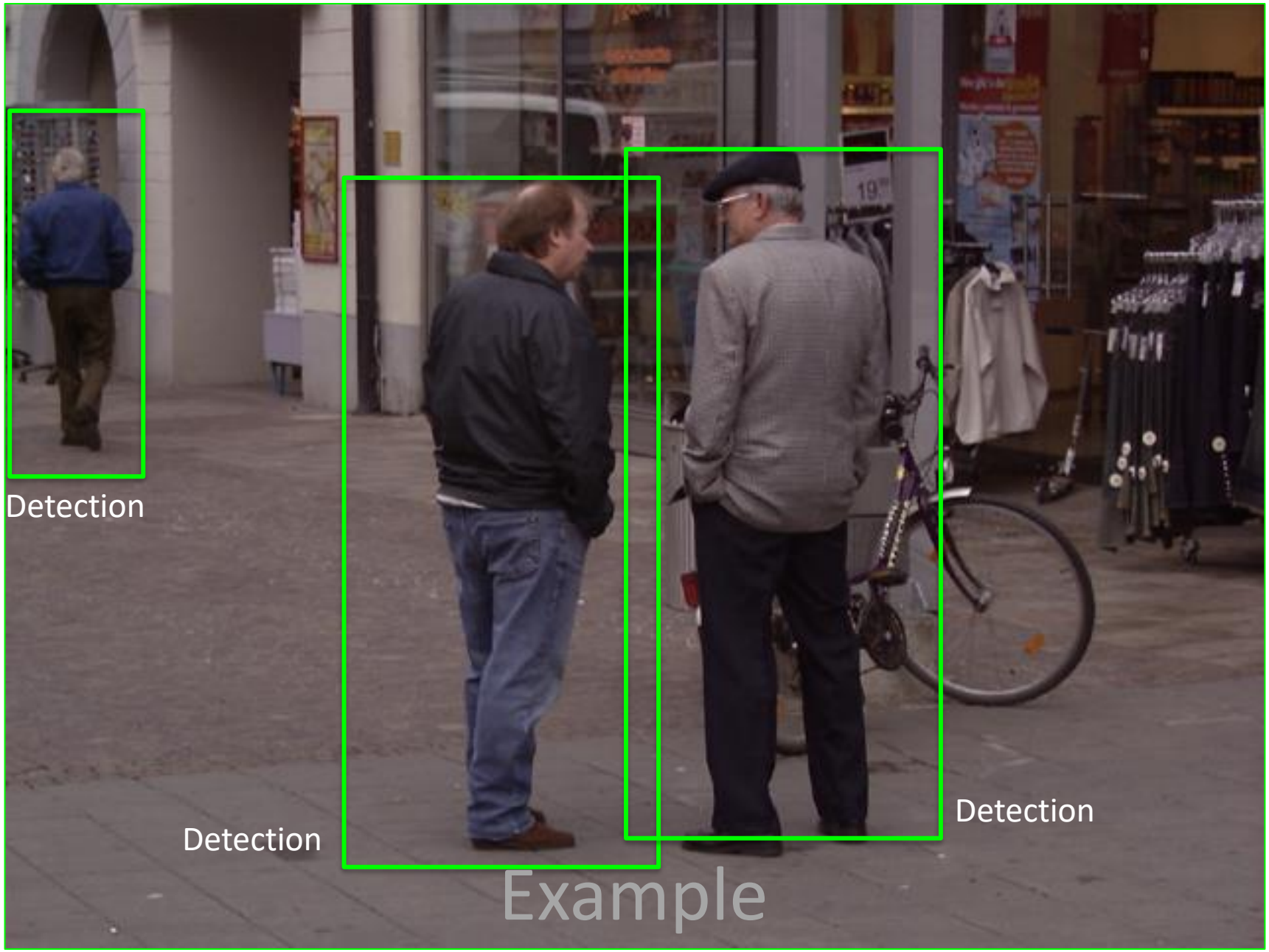

Example


Example


Detection
Example


Example


Detection
Detection
Example


Detection
Detection
Detection
Example